P1 Assignment: Dimensional Modeling

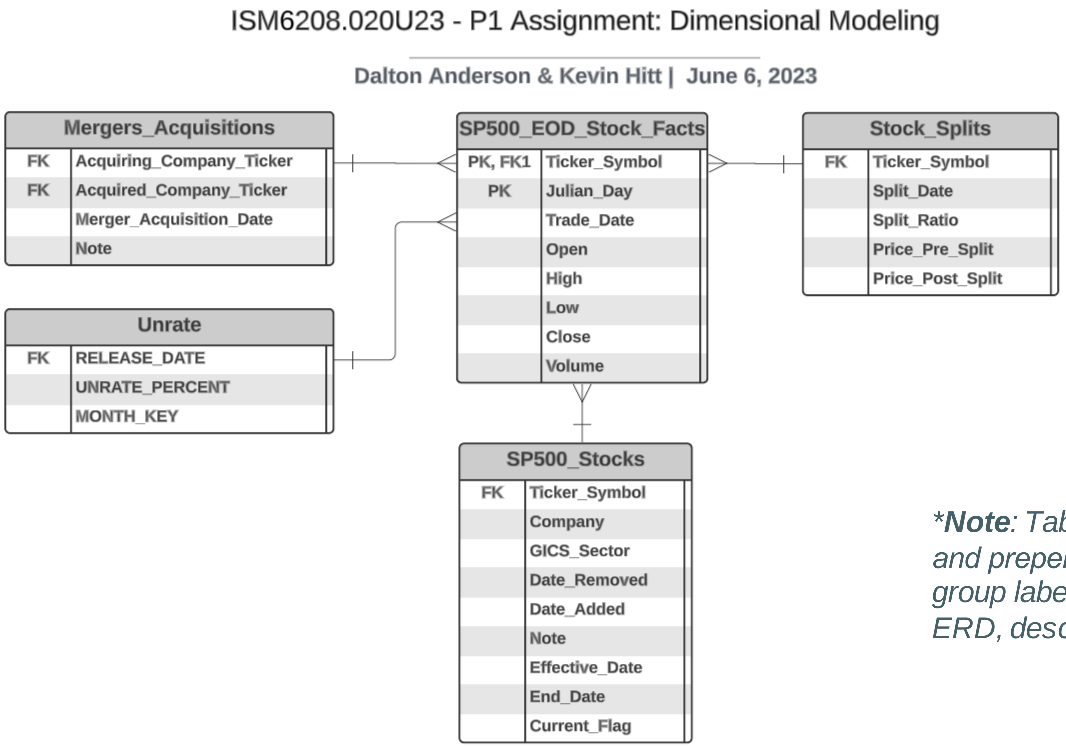
ISM6208.020U23

Dalton Anderson & Kevin Hitt | June 2023

UNIVERSITY of SOUTH FLORIDA

Part A:
Dimensional Design for Stock Market Data

Part A – Entity Relationship Diagram (ERD)



ISM6208.020U23 - P1 Assignment: Dimensional Modeling

Dalton Anderson & Kevin Hitt | June 6, 2023

	Mergers_Acquisitions
FK	Acquiring_Company_Ticker
FK	Acquired_Company_Ticker
	Merger_Acquisition_Date
	Note

	Unrate
FK	RELEASE_DATE
	UNRATE_PERCENT
	MONTH_KEY

	SP500_EOD_Stock_Facts
PK, FK1	Ticker_Symbol
PK	Julian_Day
	Trade_Date
	Open
	High
	Low
	Close
	Volume

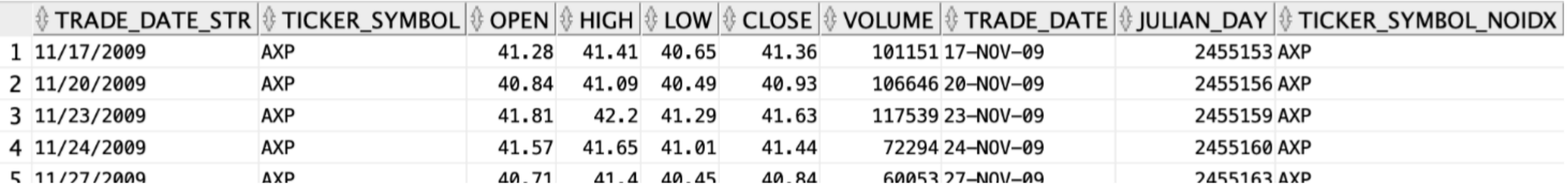
	Stock_Splits
FK	Ticker_Symbol
	Split_Date
	Split_Ratio
	Price_Pre_Split
	Price_Post_Split

	SP500_Stocks
FK	Ticker_Symbol
	Company
	GICS_Sector
	Date_Removed
	Date_Added
	Note
	Effective_Date
	End_Date
	Current_Flag

**Note*: Table capitalization and prepending of “PEARL_” group label may vary between ERD, descriptions, and queries*

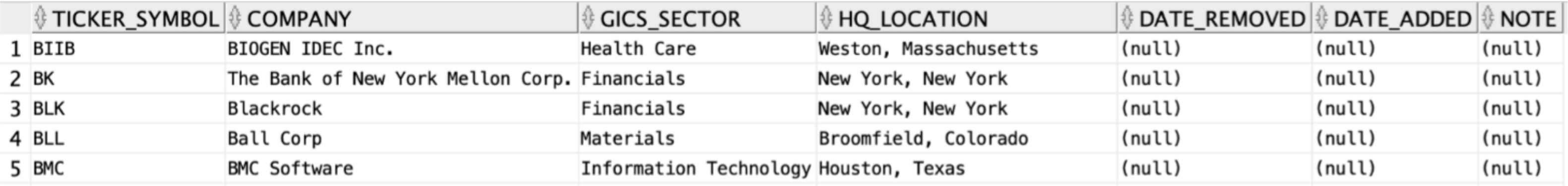
Part A – Tables Preview (1)

SP500_EOD_Stock_Facts



	TRADE_DATE_STR	TICKER_SYMBOL	OPEN	HIGH	LOW	CLOSE	VOLUME	TRADE_DATE	JULIAN_DAY	TICKER_SYMBOL_NOIDX
1	11/17/2009	AXP	41.28	41.41	40.65	41.36	101151	17-NOV-09	2455153	AXP
2	11/20/2009	AXP	40.84	41.09	40.49	40.93	106646	20-NOV-09	2455156	AXP
3	11/23/2009	AXP	41.81	42.2	41.29	41.63	117539	23-NOV-09	2455159	AXP
4	11/24/2009	AXP	41.57	41.65	41.01	41.44	72294	24-NOV-09	2455160	AXP
5	11/27/2009	AXP	40.71	41.4	40.45	40.84	60053	27-NOV-09	2455163	AXP

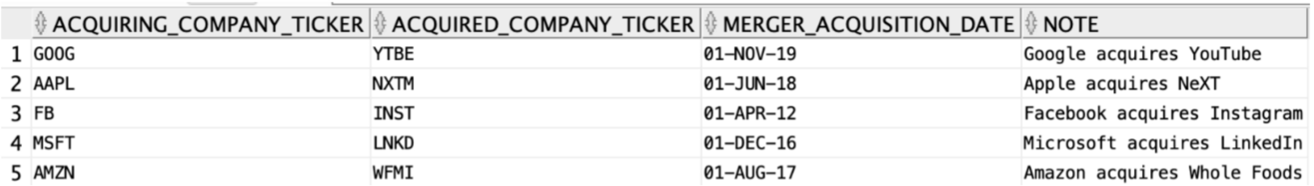
SP500_EOD_Stocks



	TICKER_SYMBOL	COMPANY	GICS_SECTOR	HQ_LOCATION	DATE_REMOVED	DATE_ADDED	NOTE
1	BIIB	BIOGEN IDEC Inc.	Health Care	Weston, Massachusetts	(null)	(null)	(null)
2	BK	The Bank of New York Mellon Corp.	Financials	New York, New York	(null)	(null)	(null)
3	BLK	Blackrock	Financials	New York, New York	(null)	(null)	(null)
4	BLL	Ball Corp	Materials	Broomfield, Colorado	(null)	(null)	(null)
5	BMC	BMC Software	Information Technology	Houston, Texas	(null)	(null)	(null)

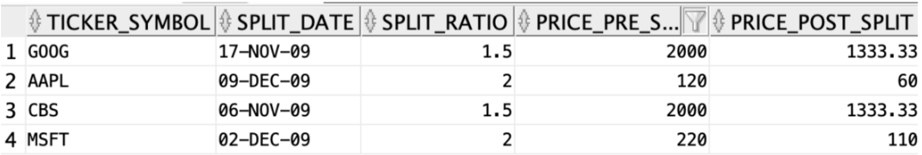
Part A – Tables Preview (2)

Mergers_Acquisitions



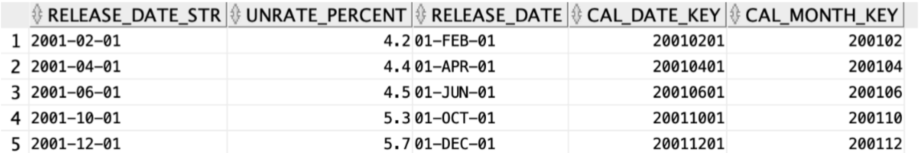
	ACQUIRING_COMPANY_TICKER	ACQUIRED_COMPANY_TICKER	MERGER_ACQUISITION_DATE	NOTE
1	GOOG	YTBE	01-NOV-19	Google acquires YouTube
2	AAPL	NXTM	01-JUN-18	Apple acquires NeXT
3	FB	INST	01-APR-12	Facebook acquires Instagram
4	MSFT	LNKD	01-DEC-16	Microsoft acquires LinkedIn
5	AMZN	WFMI	01-AUG-17	Amazon acquires Whole Foods

STOCK_SPLITS



	TICKER_SYMBOL	SPLIT_DATE	SPLIT_RATIO	PRICE_PRE_S...	PRICE_POST_SPLIT
1	GOOG	17-NOV-09	1.5	2000	1333.33
2	AAPL	09-DEC-09	2	120	60
3	CBS	06-NOV-09	1.5	2000	1333.33
4	MSFT	02-DEC-09	2	220	110

UNRATE



	RELEASE_DATE_STR	UNRATE_PERCENT	RELEASE_DATE	CAL_DATE_KEY	CAL_MONTH_KEY
1	2001-02-01	4.2	01-FEB-01	20010201	200102
2	2001-04-01	4.4	01-APR-01	20010401	200104
3	2001-06-01	4.5	01-JUN-01	20010601	200106
4	2001-10-01	5.3	01-OCT-01	20011001	200110
5	2001-12-01	5.7	01-DEC-01	20011201	200112

Part A – Design Notes

How would you extend the design to handle complexities like ***stock splits*** *(or reverse stock splits),* ***mergers and acquisitions****, or changes in business itself such as* ***new sectors****? Free to extend the project by adding any additional data.*

- **Stock_Splits** table created to track when splits occur and how they affected the stock price

- **Mergers_Acquisitions** table created to track consolidation of companies.

- *Effective_Date*, *End_Date*, *Current_Flag* added to **SP500_Stocks** table as SCD to preserve the history sectors the companies have been in over time.

- **FRED_UNRATE** table added to design (67 years monthly unemployment rate data)

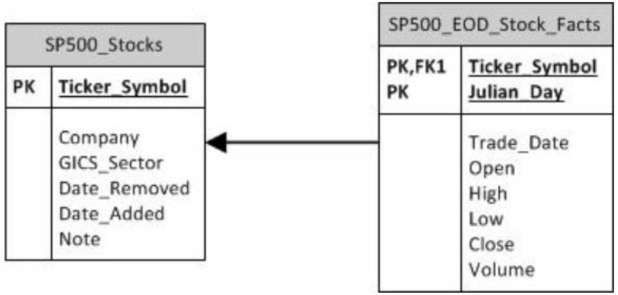
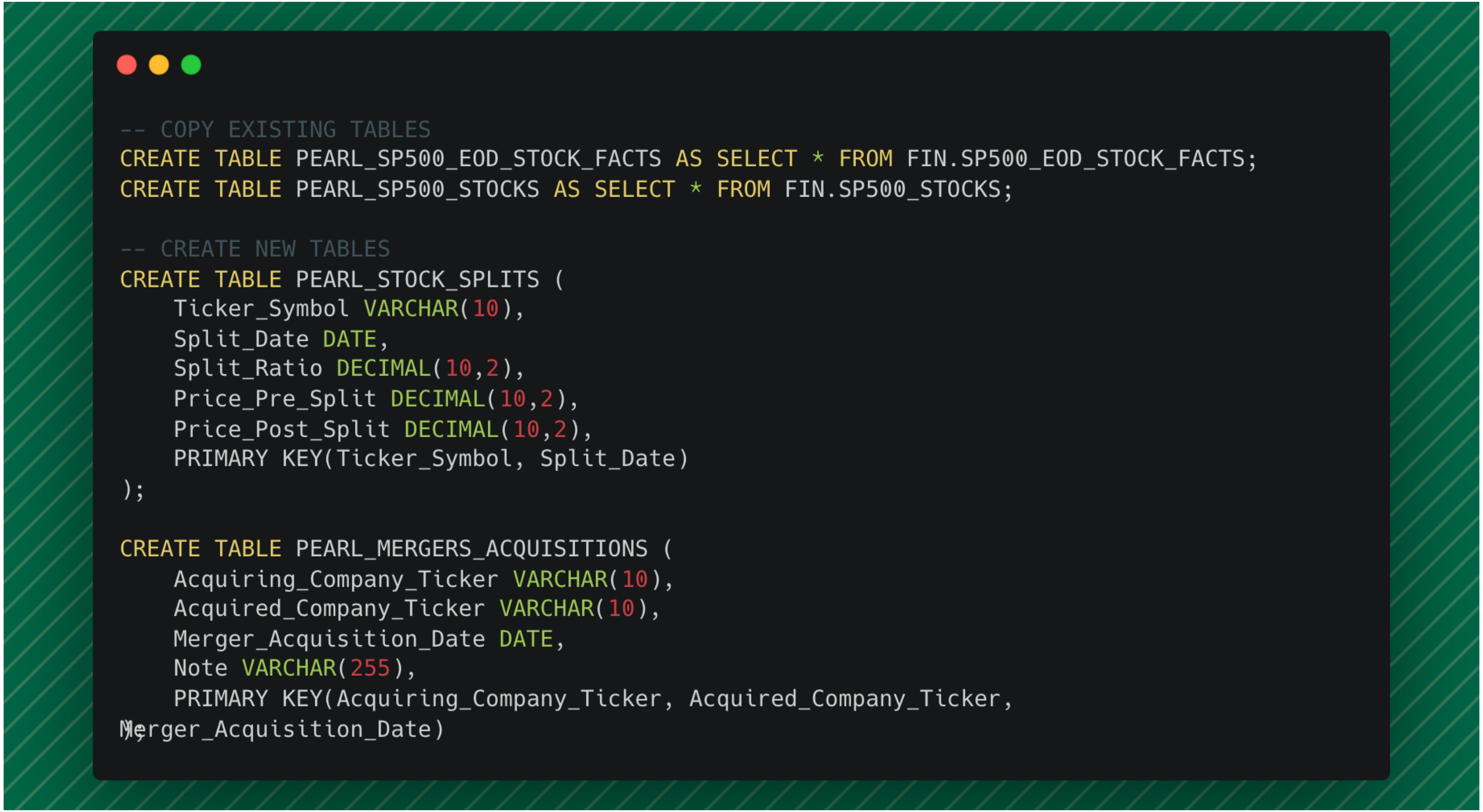
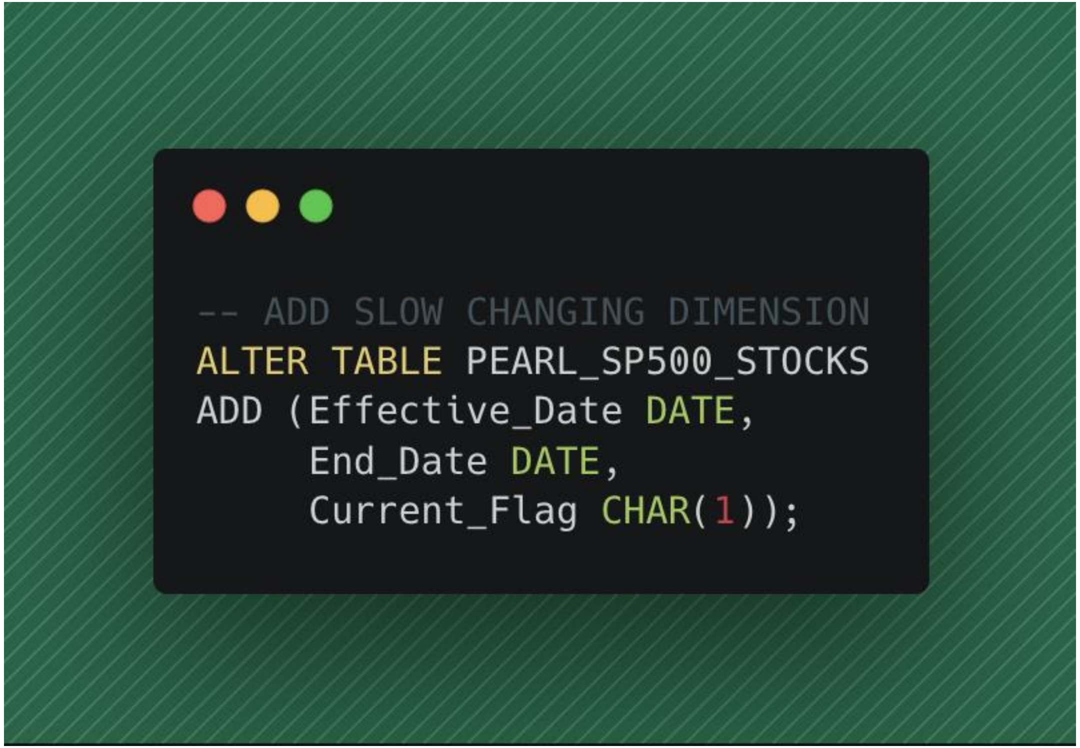


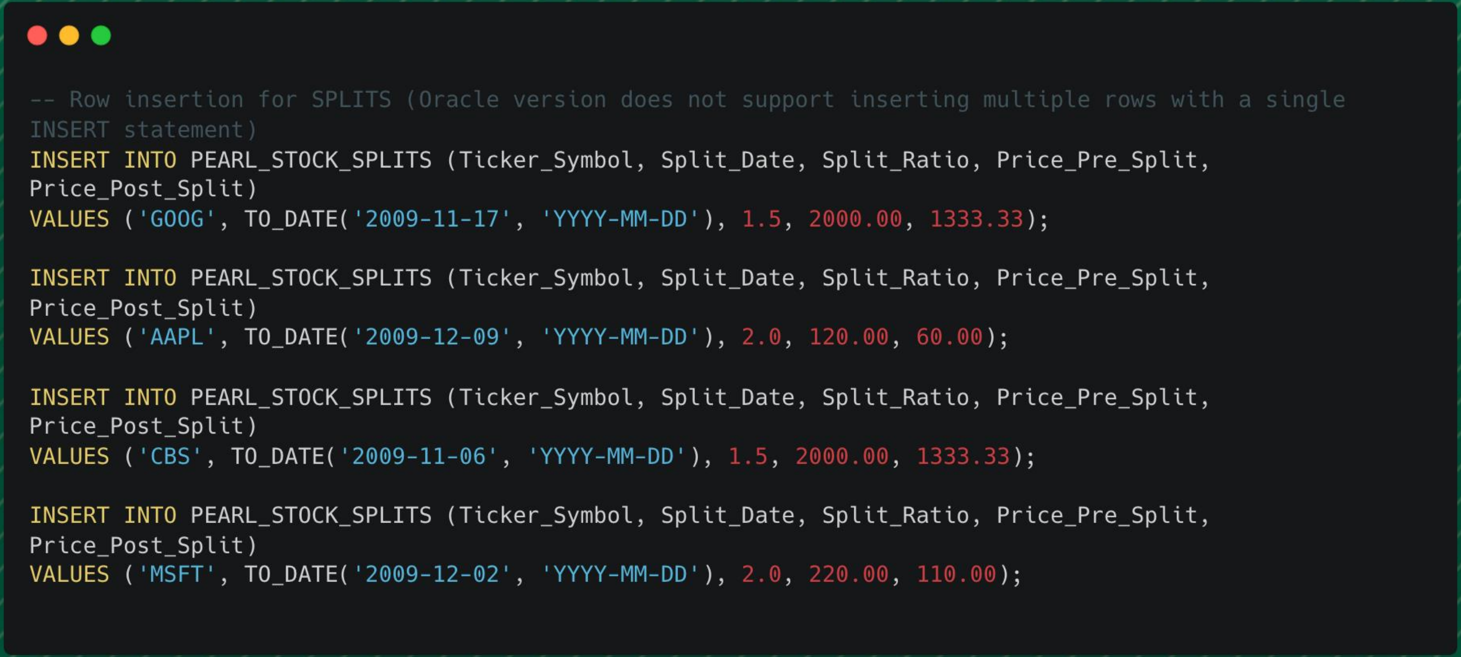
Figure 1: Base design modified



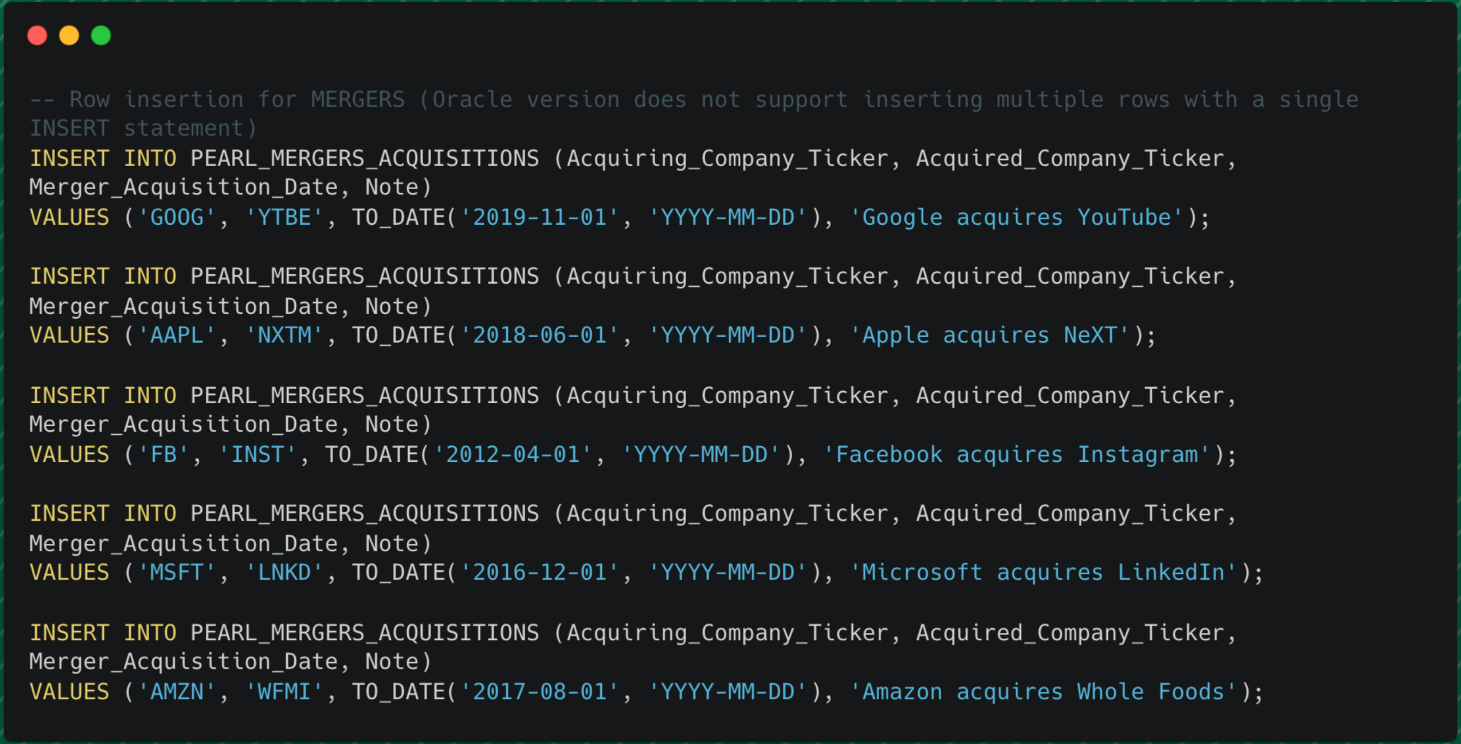
```sql
-- COPY EXISTING TABLES
CREATE TABLE PEARL_SP500_EOD_STOCK_FACTS AS SELECT * FROM FIN.SP500_EOD_STOCK_FACTS;
CREATE TABLE PEARL_SP500_STOCKS AS SELECT * FROM FIN.SP500_STOCKS;

-- CREATE NEW TABLES
CREATE TABLE PEARL_STOCK_SPLITS (
    Ticker_Symbol VARCHAR(10),
    Split_Date DATE,
    Split_Ratio DECIMAL(10,2),
    Price_Pre_Split DECIMAL(10,2),
    Price_Post_Split DECIMAL(10,2),
    PRIMARY KEY(Ticker_Symbol, Split_Date)
);

CREATE TABLE PEARL_MERGERS_ACQUISITIONS (
    Acquiring_Company_Ticker VARCHAR(10),
    Acquired_Company_Ticker VARCHAR(10),
    Merger_Acquisition_Date DATE,
    Note VARCHAR(255),
    PRIMARY KEY(Acquiring_Company_Ticker, Acquired_Company_Ticker,
Merger_Acquisition_Date)
);
```



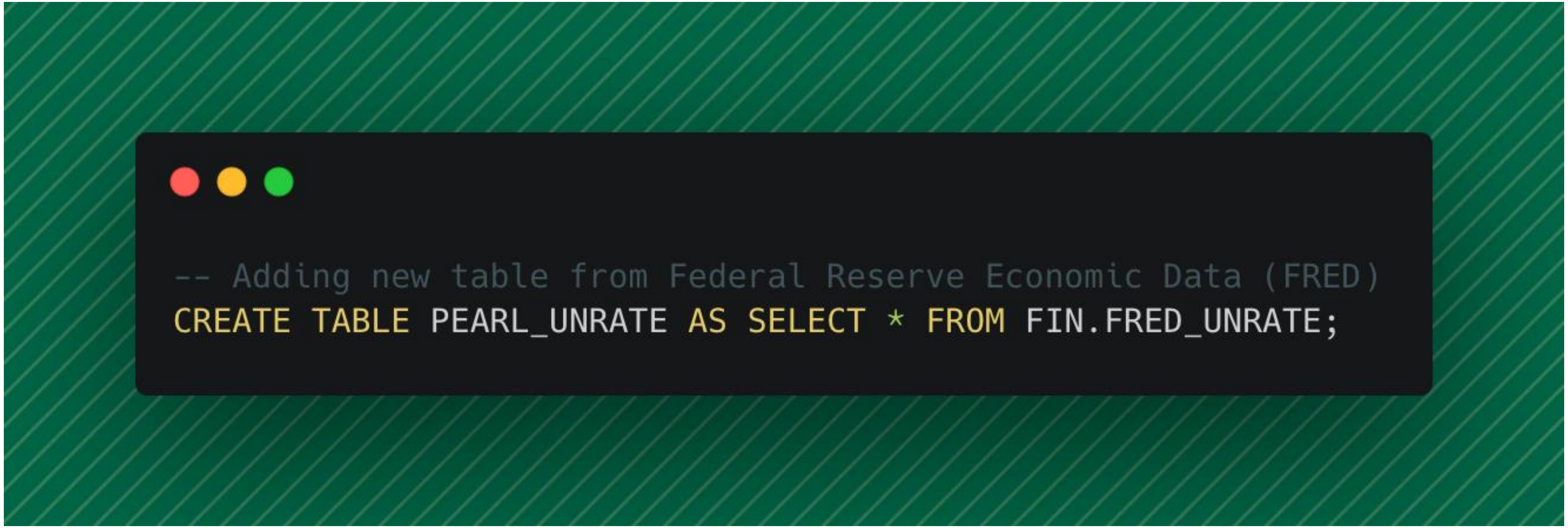
```sql
-- ADD SLOW CHANGING DIMENSION
ALTER TABLE PEARL_SP500_STOCKS
ADD (Effective_Date DATE,
     End_Date DATE,
     Current_Flag CHAR(1));
```



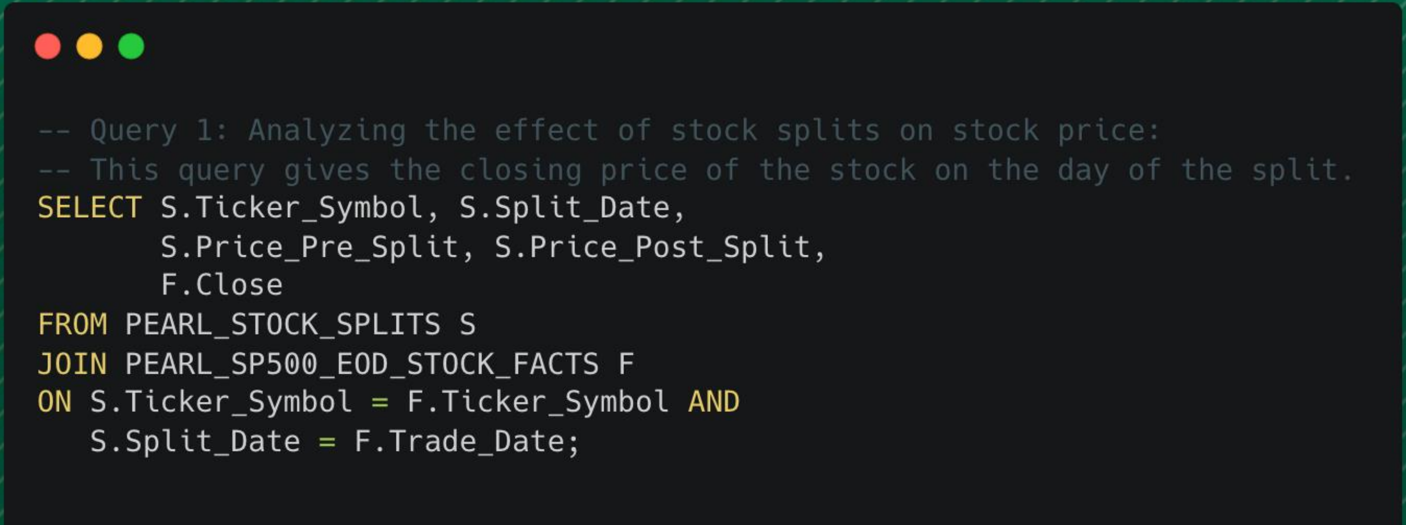
```sql
-- Row insertion for SPLITS (Oracle version does not support inserting multiple rows with a single
INSERT statement)
INSERT INTO PEARL_STOCK_SPLITS (Ticker_Symbol, Split_Date, Split_Ratio, Price_Pre_Split,
Price_Post_Split)
VALUES ('GOOG', TO_DATE('2009-11-17', 'YYYY-MM-DD'), 1.5, 2000.00, 1333.33);

INSERT INTO PEARL_STOCK_SPLITS (Ticker_Symbol, Split_Date, Split_Ratio, Price_Pre_Split,
Price_Post_Split)
VALUES ('AAPL', TO_DATE('2009-12-09', 'YYYY-MM-DD'), 2.0, 120.00, 60.00);

INSERT INTO PEARL_STOCK_SPLITS (Ticker_Symbol, Split_Date, Split_Ratio, Price_Pre_Split,
Price_Post_Split)
VALUES ('CBS', TO_DATE('2009-11-06', 'YYYY-MM-DD'), 1.5, 2000.00, 1333.33);

INSERT INTO PEARL_STOCK_SPLITS (Ticker_Symbol, Split_Date, Split_Ratio, Price_Pre_Split,
Price_Post_Split)
VALUES ('MSFT', TO_DATE('2009-12-02', 'YYYY-MM-DD'), 2.0, 220.00, 110.00);
```



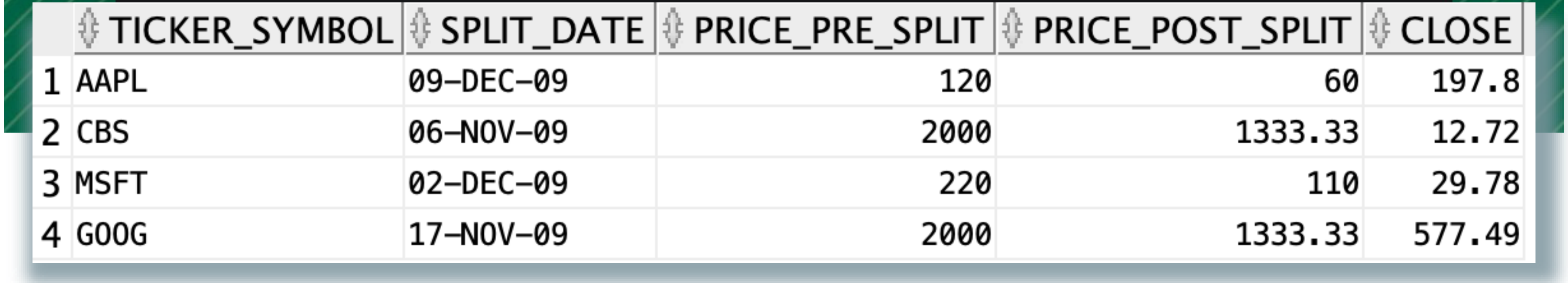
```
-- Row insertion for MERGERS (Oracle version does not support inserting multiple rows with a single
INSERT statement)
INSERT INTO PEARL_MERGERS_ACQUISITIONS (Acquiring_Company_Ticker, Acquired_Company_Ticker,
Merger_Acquisition_Date, Note)
VALUES ('GOOG', 'YTBE', TO_DATE('2019-11-01', 'YYYY-MM-DD'), 'Google acquires YouTube');

INSERT INTO PEARL_MERGERS_ACQUISITIONS (Acquiring_Company_Ticker, Acquired_Company_Ticker,
Merger_Acquisition_Date, Note)
VALUES ('AAPL', 'NXTM', TO_DATE('2018-06-01', 'YYYY-MM-DD'), 'Apple acquires NeXT');

INSERT INTO PEARL_MERGERS_ACQUISITIONS (Acquiring_Company_Ticker, Acquired_Company_Ticker,
Merger_Acquisition_Date, Note)
VALUES ('FB', 'INST', TO_DATE('2012-04-01', 'YYYY-MM-DD'), 'Facebook acquires Instagram');

INSERT INTO PEARL_MERGERS_ACQUISITIONS (Acquiring_Company_Ticker, Acquired_Company_Ticker,
Merger_Acquisition_Date, Note)
VALUES ('MSFT', 'LNKD', TO_DATE('2016-12-01', 'YYYY-MM-DD'), 'Microsoft acquires LinkedIn');

INSERT INTO PEARL_MERGERS_ACQUISITIONS (Acquiring_Company_Ticker, Acquired_Company_Ticker,
Merger_Acquisition_Date, Note)
VALUES ('AMZN', 'WFMI', TO_DATE('2017-08-01', 'YYYY-MM-DD'), 'Amazon acquires Whole Foods');
```



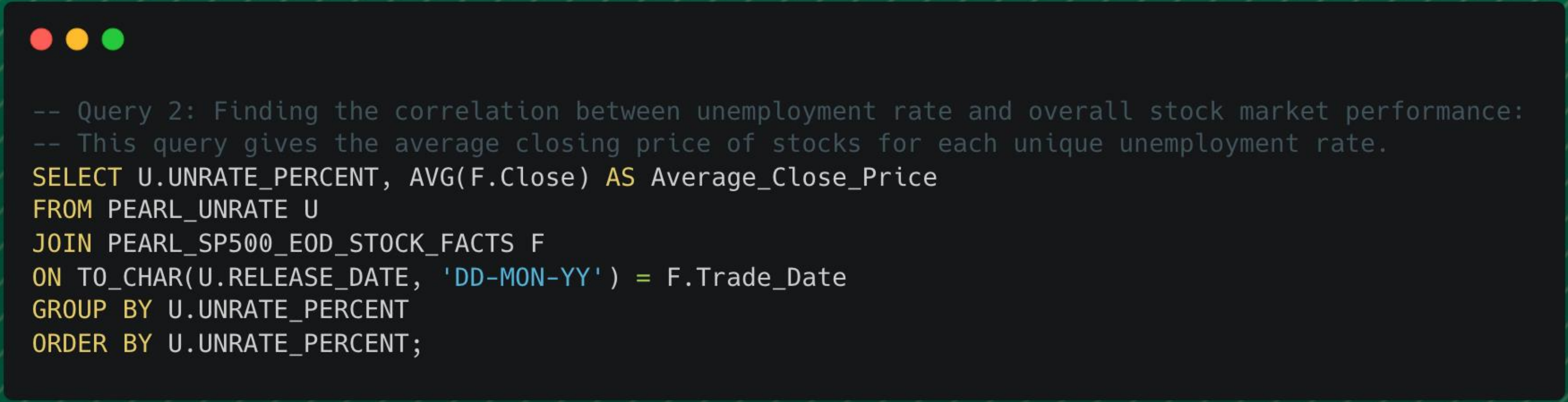
```sql
-- Adding new table from Federal Reserve Economic Data (FRED)
CREATE TABLE PEARL_UNRATE AS SELECT * FROM FIN.FRED_UNRATE;
```



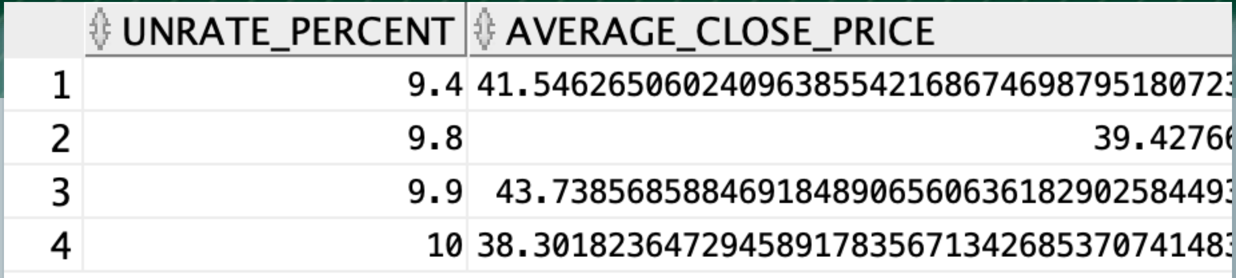
```sql
-- Query 1: Analyzing the effect of stock splits on stock price:
-- This query gives the closing price of the stock on the day of the split.
SELECT S.Ticker_Symbol, S.Split_Date,
       S.Price_Pre_Split, S.Price_Post_Split,
       F.Close
FROM PEARL_STOCK_SPLITS S
JOIN PEARL_SP500_EOD_STOCK_FACTS F
ON S.Ticker_Symbol = F.Ticker_Symbol AND
   S.Split_Date = F.Trade_Date;
```



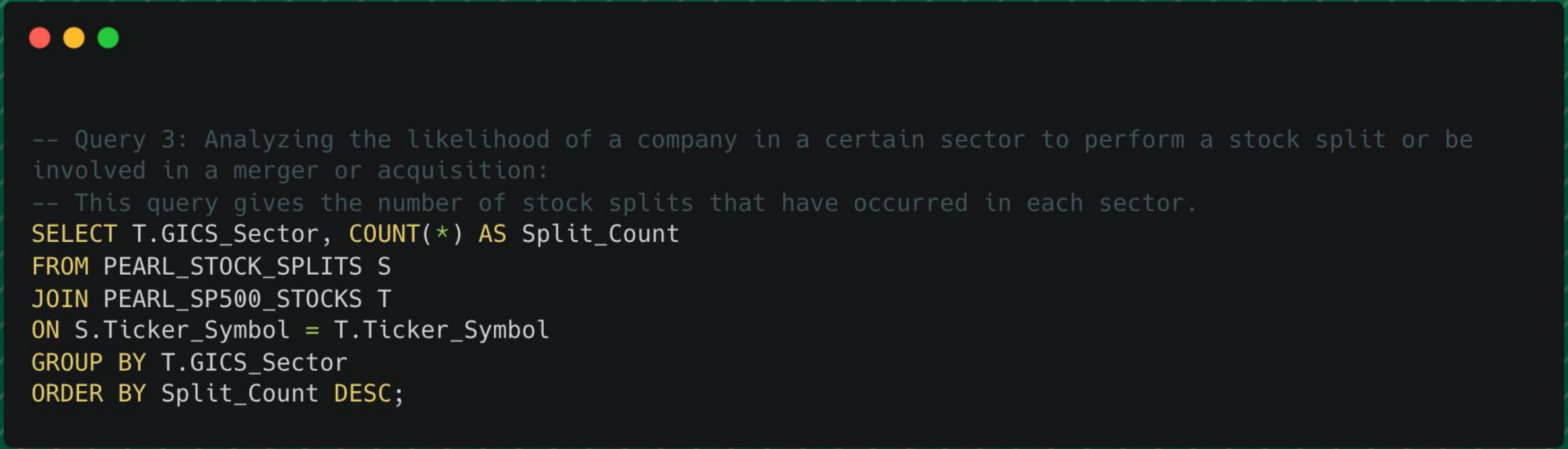
	TICKER_SYMBOL	SPLIT_DATE	PRICE_PRE_SPLIT	PRICE_POST_SPLIT	CLOSE
1	AAPL	09-DEC-09	120	60	197.8
2	CBS	06-NOV-09	2000	1333.33	12.72
3	MSFT	02-DEC-09	220	110	29.78
4	GOOG	17-NOV-09	2000	1333.33	577.49



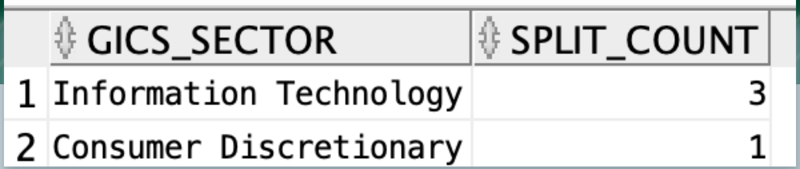
```
-- Query 2: Finding the correlation between unemployment rate and overall stock market performance:
-- This query gives the average closing price of stocks for each unique unemployment rate.
SELECT U.UNRATE_PERCENT, AVG(F.Close) AS Average_Close_Price
FROM PEARL_UNRATE U
JOIN PEARL_SP500_EOD_STOCK_FACTS F
ON TO_CHAR(U.RELEASE_DATE, 'DD-MON-YY') = F.Trade_Date
GROUP BY U.UNRATE_PERCENT
ORDER BY U.UNRATE_PERCENT;
```



	UNRATE_PERCENT	AVERAGE_CLOSE_PRICE
1	9.4	41.54626506024096385542168674698795180723
2	9.8	39.42766
3	9.9	43.7385685884691848906560636182902584493
4	10	38.30182364729458917835671342685370741483



```
-- Query 3: Analyzing the likelihood of a company in a certain sector to perform a stock split or be
involved in a merger or acquisition:
-- This query gives the number of stock splits that have occurred in each sector.
SELECT T.GICS_Sector, COUNT(*) AS Split_Count
FROM PEARL_STOCK_SPLITS S
JOIN PEARL_SP500_STOCKS T
ON S.Ticker_Symbol = T.Ticker_Symbol
GROUP BY T.GICS_Sector
ORDER BY Split_Count DESC;
```

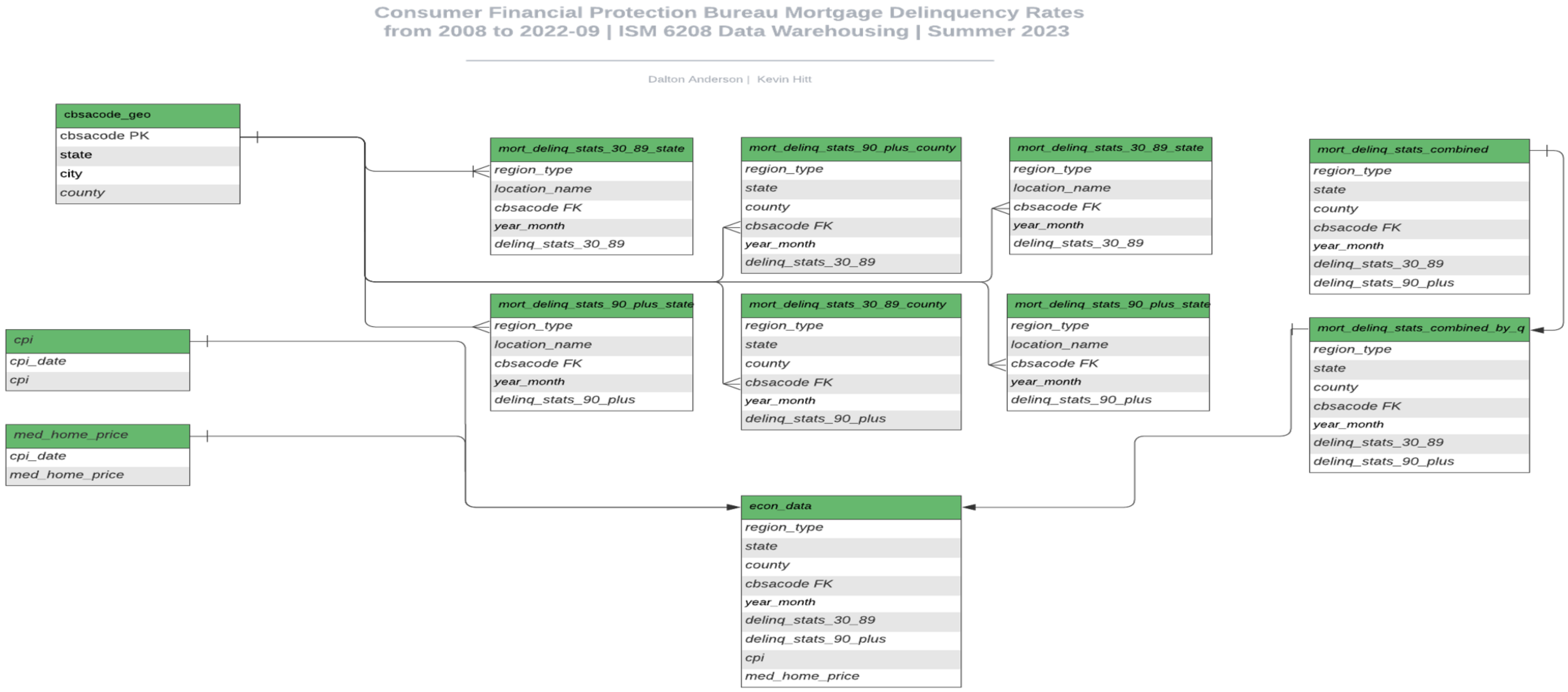


	GICS_SECTOR	SPLIT_COUNT
1	Information Technology	3
2	Consumer Discretionary	1

Part B:
Dimensional Model for Data of Your Choice

Part B – Consumer Financial Protection Bureau Mortgage Delinquency Rates from 2008 to 2022-09

Part B – Entity Relationship Diagram (ERD) (1)



Consumer Financial Protection Bureau Mortgage Delinquency Rates
from 2008 to 2022-09 | ISM 6208 Data Warehousing | Summer 2023

Dalton Anderson | Kevin Hitt

cbsacode_geo
cbsacode PK
state
city
county

mort_delinq_stats_30_89_state
region_type
location_name
cbsacode FK
year_month
delinq_stats_30_89

mort_delinq_stats_90_plus_county
region_type
state
county
cbsacode FK
year_month
delinq_stats_30_89

mort_delinq_stats_30_89_state
region_type
location_name
cbsacode FK
year_month
delinq_stats_30_89

mort_delinq_stats_combined
region_type
state
county
cbsacode FK
year_month
delinq_stats_30_89
delinq_stats_90_plus

mort_delinq_stats_90_plus_state
region_type
location_name
cbsacode FK
year_month
delinq_stats_90_plus

mort_delinq_stats_30_89_county
region_type
state
county
cbsacode FK
year_month
delinq_stats_90_plus

mort_delinq_stats_90_plus_state
region_type
location_name
cbsacode FK
year_month
delinq_stats_90_plus

cpi
cpi_date
cpi

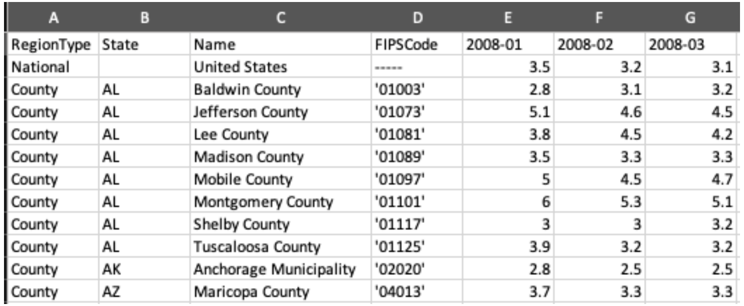
med_home_price
cpi_date
med_home_price

mort_delinq_stats_combined_by_q
region_type
state
county
cbsacode FK
year_month
delinq_stats_30_89
delinq_stats_90_plus

econ_data
region_type
state
county
cbsacode FK
year_month
delinq_stats_30_89
delinq_stats_90_plus
cpi
med_home_price

Part B – Tables Preview ETL (2)

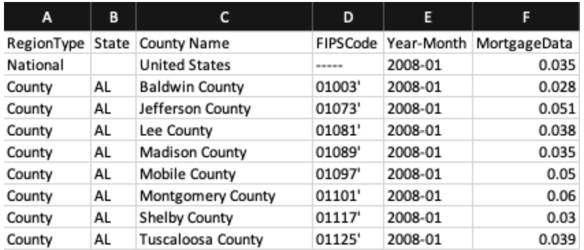
CountyMortgagesPercent-30-89Day



A	B	C	D	E	F	G
RegionType	State	Name	FIPSCode	2008-01	2008-02	2008-03
National		United States	-----	3.5	3.2	3.1
County	AL	Baldwin County	'01003'	2.8	3.1	3.2
County	AL	Jefferson County	'01073'	5.1	4.6	4.5
County	AL	Lee County	'01081'	3.8	4.5	4.2
County	AL	Madison County	'01089'	3.5	3.3	3.3
County	AL	Mobile County	'01097'	5	4.5	4.7
County	AL	Montgomery County	'01101'	6	5.3	5.1
County	AL	Shelby County	'01117'	3	3	3.2
County	AL	Tuscaloosa County	'01125'	3.9	3.2	3.2
County	AK	Anchorage Municipality	'02020'	2.8	2.5	2.5
County	AZ	Maricopa County	'04013'	3.7	3.3	3.3

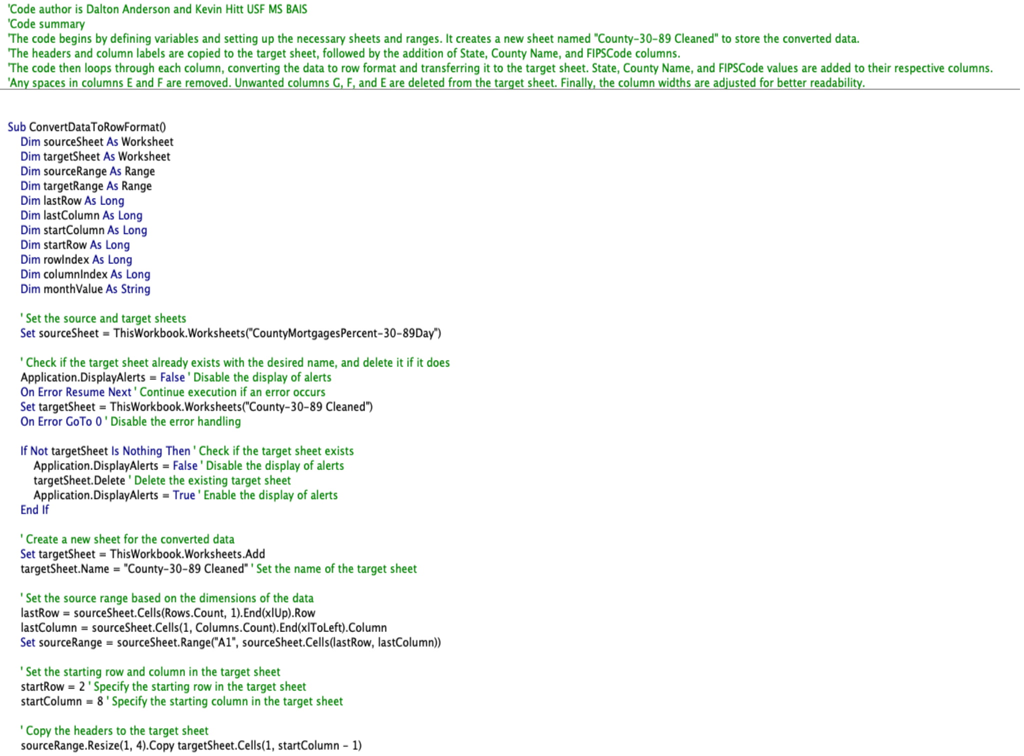
**Note: The majority of time in Part B was dedicated to the ETL process (Extract, Transform, Load). This process entailed constructing a macro for each state, county, and metropolitan area file to change the date format from column-based to row-based.*

County-30-89 Cleaned

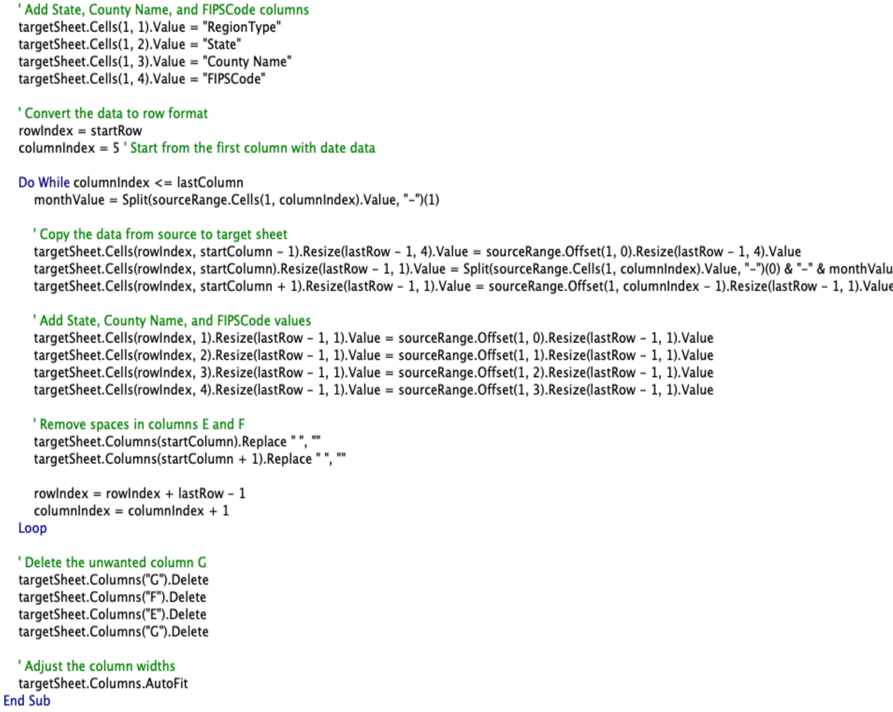


A	B	C	D	E	F
RegionType	State	County Name	FIPSCode	Year-Month	MortgageData
National		United States	-----	2008-01	0.035
County	AL	Baldwin County	01003'	2008-01	0.028
County	AL	Jefferson County	01073'	2008-01	0.051
County	AL	Lee County	01081'	2008-01	0.038
County	AL	Madison County	01089'	2008-01	0.035
County	AL	Mobile County	01097'	2008-01	0.05
County	AL	Montgomery County	01101'	2008-01	0.06
County	AL	Shelby County	01117'	2008-01	0.03
County	AL	Tuscaloosa County	01125'	2008-01	0.039

Part B – Tables Preview ETL Macro (3)

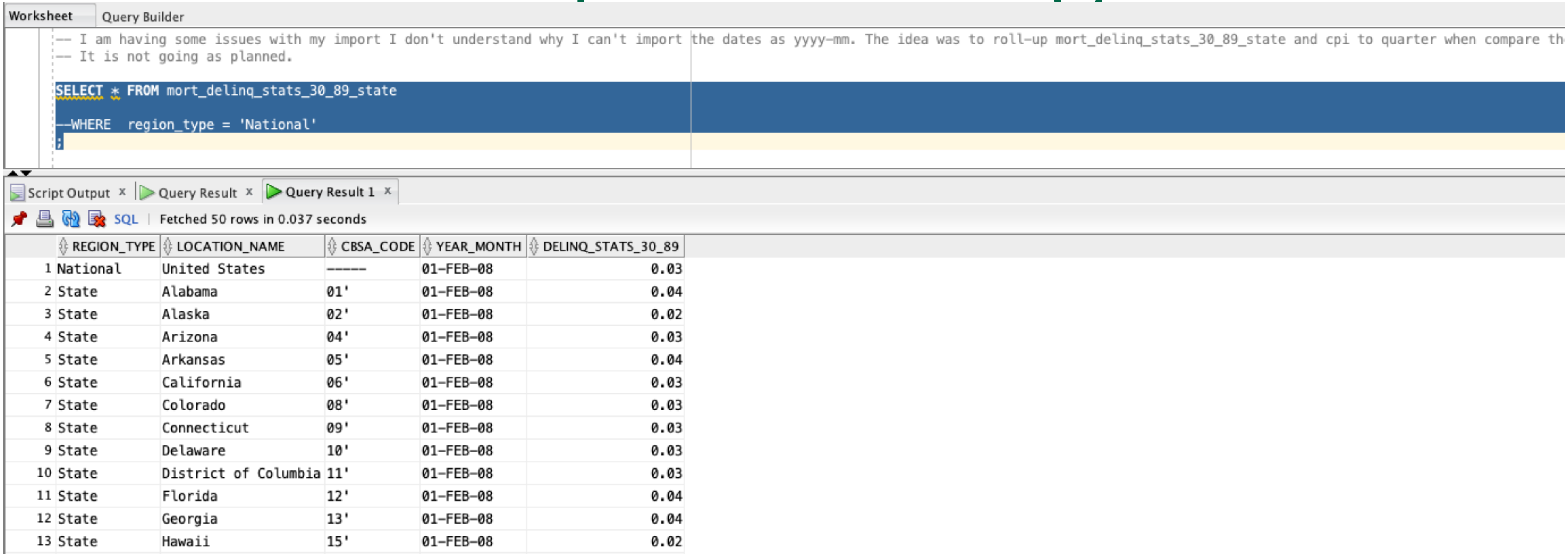


```
'Code author is Dalton Anderson and Kevin Hitt USF MS BAIS
'Code summary
'The code begins by defining variables and setting up the necessary sheets and ranges. It creates a new sheet named "County-30-89 Cleaned" to store the converted data.
'The headers and column labels are copied to the target sheet, followed by the addition of State, County Name, and FIPSCode columns.
'The code then loops through each column, converting the data to row format and transferring it to the target sheet. State, County Name, and FIPSCode values are added to their respective columns.
'Any spaces in columns E and F are removed. Unwanted columns G, F, and E are deleted from the target sheet. Finally, the column widths are adjusted for better readability.


Sub ConvertDataToRowFormat()
    Dim sourceSheet As Worksheet
    Dim targetSheet As Worksheet
    Dim sourceRange As Range
    Dim targetRange As Range
    Dim lastRow As Long
    Dim lastColumn As Long
    Dim startColumn As Long
    Dim startRow As Long
    Dim rowIndex As Long
    Dim columnIndex As Long
    Dim monthValue As String

    ' Set the source and target sheets
    Set sourceSheet = ThisWorkbook.Worksheets("CountyMortgagesPercent-30-89Day")

    ' Check if the target sheet already exists with the desired name, and delete it if it does
    Application.DisplayAlerts = False ' Disable the display of alerts
    On Error Resume Next ' Continue execution if an error occurs
    Set targetSheet = ThisWorkbook.Worksheets("County-30-89 Cleaned")
    On Error GoTo 0 ' Disable the error handling

    If Not targetSheet Is Nothing Then ' Check if the target sheet exists
        Application.DisplayAlerts = False ' Disable the display of alerts
        targetSheet.Delete ' Delete the existing target sheet
        Application.DisplayAlerts = True ' Enable the display of alerts
    End If

    ' Create a new sheet for the converted data
    Set targetSheet = ThisWorkbook.Worksheets.Add
    targetSheet.Name = "County-30-89 Cleaned" ' Set the name of the target sheet

    ' Set the source range based on the dimensions of the data
    lastRow = sourceSheet.Cells(Rows.Count, 1).End(xlUp).Row
    lastColumn = sourceSheet.Cells(1, Columns.Count).End(xlToLeft).Column
    Set sourceRange = sourceSheet.Range("A1", sourceSheet.Cells(lastRow, lastColumn))

    ' Set the starting row and column in the target sheet
    startRow = 2 ' Specify the starting row in the target sheet
    startColumn = 8 ' Specify the starting column in the target sheet

    ' Copy the headers to the target sheet
    sourceRange.Resize(1, 4).Copy targetSheet.Cells(1, startColumn - 1)
```



```
    ' Add State, County Name, and FIPSCode columns
    targetSheet.Cells(1, 1).Value = "RegionType"
    targetSheet.Cells(1, 2).Value = "State"
    targetSheet.Cells(1, 3).Value = "County Name"
    targetSheet.Cells(1, 4).Value = "FIPSCode"

    ' Convert the data to row format
    rowIndex = startRow
    columnIndex = 5 ' Start from the first column with date data

    Do While columnIndex <= lastColumn
        monthValue = Split(sourceRange.Cells(1, columnIndex).Value, "-")(1)

        ' Copy the data from source to target sheet
        targetSheet.Cells(rowIndex, startColumn - 1).Resize(lastRow - 1, 4).Value = sourceRange.Offset(1, 0).Resize(lastRow - 1, 4).Value
        targetSheet.Cells(rowIndex, startColumn).Resize(lastRow - 1, 1).Value = Split(sourceRange.Cells(1, columnIndex).Value, "-")(0) & "-" & monthValu
        targetSheet.Cells(rowIndex, startColumn + 1).Resize(lastRow - 1, 1).Value = sourceRange.Offset(1, columnIndex - 1).Resize(lastRow - 1, 1).Value

        ' Add State, County Name, and FIPSCode values
        targetSheet.Cells(rowIndex, 1).Resize(lastRow - 1, 1).Value = sourceRange.Offset(1, 0).Resize(lastRow - 1, 1).Value
        targetSheet.Cells(rowIndex, 2).Resize(lastRow - 1, 1).Value = sourceRange.Offset(1, 1).Resize(lastRow - 1, 1).Value
        targetSheet.Cells(rowIndex, 3).Resize(lastRow - 1, 1).Value = sourceRange.Offset(1, 2).Resize(lastRow - 1, 1).Value
        targetSheet.Cells(rowIndex, 4).Resize(lastRow - 1, 1).Value = sourceRange.Offset(1, 3).Resize(lastRow - 1, 1).Value

        ' Remove spaces in columns E and F
        targetSheet.Columns(startColumn).Replace " ", ""
        targetSheet.Columns(startColumn + 1).Replace " ", ""

        rowIndex = rowIndex + lastRow - 1
        columnIndex = columnIndex + 1
    Loop

    ' Delete the unwanted column G
    targetSheet.Columns("G").Delete
    targetSheet.Columns("F").Delete
    targetSheet.Columns("E").Delete
    targetSheet.Columns("G").Delete

    ' Adjust the column widths
    targetSheet.Columns.AutoFit
End Sub
```

Part B – Tables
Preview mort_delinq_stats_30_89_state (4)

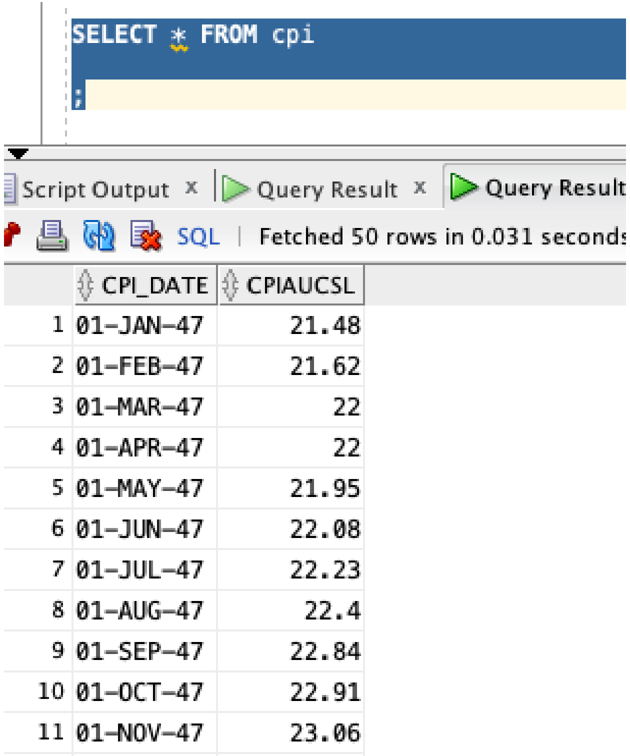


```
-- I am having some issues with my import I don't understand why I can't import the dates as yyyy-mm. The idea was to roll-up mort_delinq_stats_30_89_state and cpi to quarter when compare th
-- It is not going as planned.

SELECT * FROM mort_delinq_stats_30_89_state

--WHERE  region_type = 'National'
;
```

Fetched 50 rows in 0.037 seconds

	REGION_TYPE	LOCATION_NAME	CBSA_CODE	YEAR_MONTH	DELINQ_STATS_30_89
1	National	United States	------	01-FEB-08	0.03
2	State	Alabama	01'	01-FEB-08	0.04
3	State	Alaska	02'	01-FEB-08	0.02
4	State	Arizona	04'	01-FEB-08	0.03
5	State	Arkansas	05'	01-FEB-08	0.04
6	State	California	06'	01-FEB-08	0.03
7	State	Colorado	08'	01-FEB-08	0.03
8	State	Connecticut	09'	01-FEB-08	0.03
9	State	Delaware	10'	01-FEB-08	0.03
10	State	District of Columbia	11'	01-FEB-08	0.03
11	State	Florida	12'	01-FEB-08	0.04
12	State	Georgia	13'	01-FEB-08	0.04
13	State	Hawaii	15'	01-FEB-08	0.02

Part B – Tables Preview CPI (5)

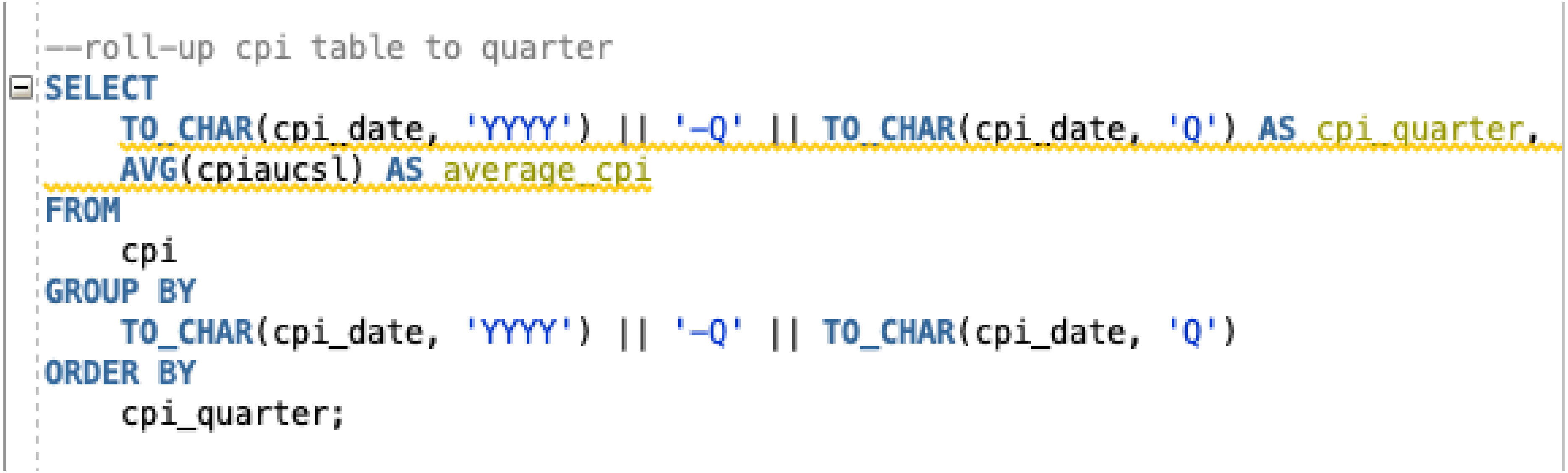


```
SELECT * FROM cpi
;
```

Fetched 50 rows in 0.031 seconds

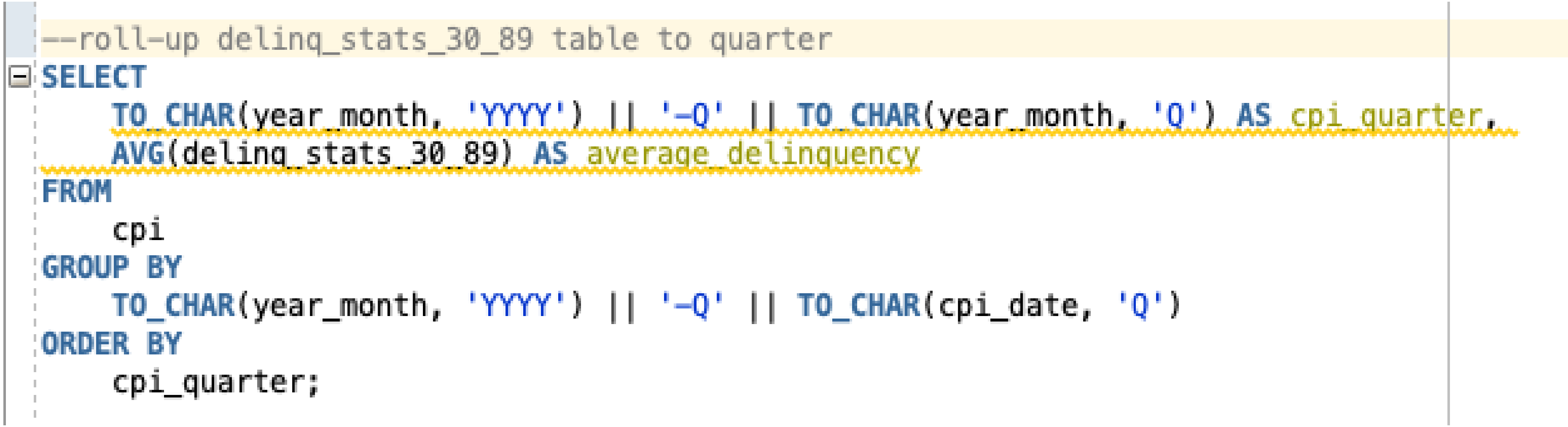
	CPI_DATE	CPIAUCSL
1	01-JAN-47	21.48
2	01-FEB-47	21.62
3	01-MAR-47	22
4	01-APR-47	22
5	01-MAY-47	21.95
6	01-JUN-47	22.08
7	01-JUL-47	22.23
8	01-AUG-47	22.4
9	01-SEP-47	22.84
10	01-OCT-47	22.91
11	01-NOV-47	23.06

Part B – SQL Preview CPI Roll-up



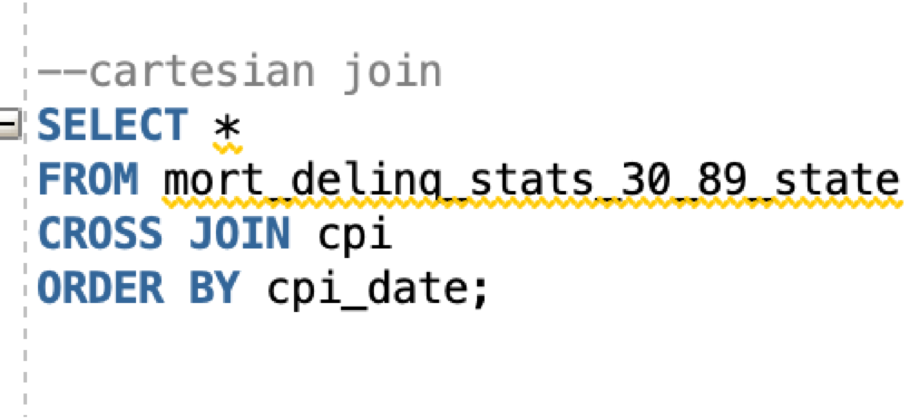
```
--roll-up cpi table to quarter
SELECT
    TO_CHAR(cpi_date, 'YYYY') || '-Q' || TO_CHAR(cpi_date, 'Q') AS cpi_quarter,
    AVG(cpiaucsl) AS average_cpi
FROM
    cpi
GROUP BY
    TO_CHAR(cpi_date, 'YYYY') || '-Q' || TO_CHAR(cpi_date, 'Q')
ORDER BY
    cpi_quarter;
```

Part B – SQL Preview mort_delinq_stats_30_89_state Roll-up



```
--roll-up delinq_stats_30_89 table to quarter
SELECT
    TO_CHAR(year_month, 'YYYY') || '-Q' || TO_CHAR(year_month, 'Q') AS cpi_quarter,
    AVG(delinq_stats_30_89) AS average_delinquency
FROM
    cpi
GROUP BY
    TO_CHAR(year_month, 'YYYY') || '-Q' || TO_CHAR(cpi_date, 'Q')
ORDER BY
    cpi_quarter;
```

Part B – SQL Preview Cartesian join



```
--cartesian join
SELECT *
FROM mort_delinq_stats_30_89_state
CROSS JOIN cpi
ORDER BY cpi_date;
```

Thank you!

GITHUB LINK

anderson669@usf.edu
kevinhitt@usf.edu

UNIVERSITY of SOUTH FLORIDA